## **APPENDIX**

The following code listing shows one implementation of the conventional VSCALE routine in accordance with ITU (International Telecommunication Union)-T Recommendation G.728 – Annex G.

5

```
; search for maximum positive and negative values in vector
                                movs.w @r4+,y0
             pcopy y0,y1.
                                movx.w @r4+,x1
 10
             pcmp x1,y0
         dct pcopy x1,y0
             pcmp x1,y1
         dcf pcopy x1,y1
                                 movx.w @r4+,x1
             pcmp x1,y0
 15
         dct pcopy x1,y0
             pcmp x1,y1
         dcf pcopy x1,y1
                                 movx.w @r4+,x1
             pcmp x1,y0
         dct pcopy x1,y0
20
0
             pcmp x1,y1
         dcf pcopy x1,y1
                                 movx.w @r4,x1
             pcmp x1,y0
                                 movx.w @r4+r8,x0
Ħ
         dct pcopy x1,y0
                                 movx.w @r4+r8,x0
H
             pcmp x1,y1
                                 movx.w @r4+r8,x0
25
         dcf pcopy x1,y1
                                 movx.w @r4+r8,x0
إيوا
Ħ
             sts
                   y0,r1
      mov
             r1,r0
ij
             sts
                   y1,r7
30 mm sum mm mm 35
             or
                   r7,r0
                   r0,r0
             tst
                   VS_ZERO
             bt
             pabs
                   y1,y1
             pclr
                   a0
             pinc
                   a0,a0
             lds
                   r6,y0
             psha
                   #16,y0
             psha
                   a0,y0,a0
 40
             sts
                   y1,r0
                         r0,r1
             cmp/ge
             bt/s
                   vs pos
                    #0,r0
             mov
 45
             sts
                   a0, r3
                   r3, r3
             neg
             mov
                   r3, r2
             shll
                   r2
 50
             cmp/ge
                          r2, r7
             bf
                    vsloop3
             cmp/gt
                          r7, r3
             bt
                   vs_end2
 55
       ; Case 3: maximum negative value still has room for normalization
             .align
       vsloop41:
             shal r7
 60
             cmp/gt
                          r7, r3
             bf/s vsloop41
             add #1,r0
```

```
lds
                  r0,y0
      psha #16,y0
            movs.w @r4+,x1
  5
            psha x1,y0,a0
                               movx.w @r4+,x1
      psha x1,y0,a1
                      movx.w @r4+,x1
                               movs.w a0,@r5+
                               movx.w a1,@r5+
                               movx.w @r4+,x1
            psha x1,y0,a0
 10
      psha x1,y0,a1
                         movx.w @r4+,x1
                               movx.w a0,@r5+
            psha x1,y0,a0
      movx.w a1,@r5+
                               movx.w a0,@r5+
 15
            rts
            nop
      ; Case 2: maximum negative value exceeds minimum range vsloop3:
 20
                       r2,r7
            cmp/ge
            bt
                  vs_end2
13
             .align
i Li
      vsloop31:
IJ
            shar r7
25
            cmp/ge
                         r2, r7
Ţ,
      bf/s vsloop31
144
            add
                  #-1,r0
lds
                  r0,y0
            psha #16,y0
            movs.w @r4+,x1
psha x1,y0,a0
psha x1,y0,a1
                               movx.w @r4+,x1
ij.
                               movx.w @r4+,x1
1
                               movs.w a0,@r5+
35
                               movx.w a1,@r5+
psha x1,y0,a0
                               movx.w @r4+,x1
1 =
            psha x1,y0,a1
                               movx.w @r4+,x1
                               movx.w a0,@r5+
            psha x1,y0,a0
 40
                               movx.w a1,@r5+
                               movx.w a0,@r5+
            rts
            nop
 45
      ; Case 1: zero input vector
      VS_ZER0:
            pclr a0
                               movs.w a0,@r5+
 50
                               movx.w a0,@r5+
                               movx.w a0,@r5+
                               movx.w a0,@r5+
                               movx.w a0,@r5+
            mov r6, r0
 55
             add #1,r0
             rts
             nop
 60
             .align
```

vs\_pos:

```
a0,r2
                                        sts
                                                            r2, r3
                                        mov
                                         #-1,r3
                     add
                                         add
                                                            r2,r3
       5
                                                            r1, r3
                     cmp/ge
                                        bf
                                                             vsloop5
                     cmp/ge
                                                             r2, r1
    10
                                                             vs end2
                                        bt
                     ; Case 5: maximum positive value still has room for normalization
                                          .aliqn
                     vsloop61:
    15
                                          shal r1
                                          cmp/ge
                                                                                  r2,r1
                                         bf/s vsloop61
                                         add #1,r0
                     vs_end2:
    20
                                                             r0,y0
                                          lds
                                          psha #16,y0
  1
                                          movs.w @r4+,x1
  1
                                          psha x1,y0,a0
                                                                                                      movx.w @r4+,x1
  H
                                                                                                      movx.w @r4+,x1
                                          psha x1,y0,a1
                                                                                                      movs.w a0,@r5+
25
                                                                                                      movx.w a1,@r5+
  movx.w @r4+,x1
                                          psha x1,y0,a0
  ٠....
                                                                                                       movx.w @r4+,x1
                                          psha x1,y0,a1
  M
                                                                                                       movx.w a0,@r5+
  30
                                          psha x1,y0,a0
                                                                                                       movx.w a1,@r5+
  The state of the s
                                                                                                       movx.w a0,@r5+
  rts
  35
                                          nop
  ; Case 4: maximum positive value exceeds maximum range
                       vsloop5
                                                                                    r1, r3
                                           cmp/ge
      40
                                          bt
                                                               vs end2
                                            .align
                       vsloop5:
                                           shar rl
      45
                                           cmp/qe
                                                                                    r1, r3
                                                             vsloop51
                                           bf/s
                                           add
                                                               #-1,r0
                                           bra
                                                                vs_end2
       50
                                           nop
                        The following is an algorithm in accordance with a first embodiment of the
                        present invention.
       55
                        ; search for minimum NLS
                                                                                                         movs.w @r4+,x0
                                                                                                         movx.w @r4+,x0
                                            pdmsb x0,a0
                                            pdmsb x0,y0
       60
                                            pcmp a0,y0
                                                                                                         movx.w @r4+,x0
                               dct pcopy y0,a0
```

```
pdmsb x0,y0
           pcmp a0,y0
                              movx.w @r4+,x0
       dct pcopy y0,a0
            pdmsb x0,y0
 5
           pcmp a0,y0
                                     movx.w @r4,x0
                 pcopy y0,a0
       dct
                              movx.w @r4+r8,x1;dummy movx to reset r4=&IN[0]
            pdmsb x0,y0
                              movx.w @r4+r8,x1
            pcmp a0,y0
                              movx.w @r4+r8,x1
       dct pcopy y0,a0
                              movx.w @r4+r8,x1
10
            psha #-16,a0
                                     ;r0=NLS MIN
                  a0, r0
            sts
      ;Case 1: zero input vector
15
                     #31, r0
            cmp/eq
            bf/s VSCALE1_check_NLSeq31_end
                              ;r6=MLS
                  r6, r7
            mov
                  r6, r0
            mov
                                     ; set r0=NLS = MLS + 1
20
                  #1, r0
            add
            pclr a0
                               movs.w a0,@r5+
ı.
                               movx.w a0,@r5+
movx.w a0,@r5+
                               movx.w a0,@r5+
25
                               movx.w a0,@r5+
111
            rts
1.1
            nop
Ħ
      ; Case 2: non-zero input vector
      VSCALE1 check_NLSeq31_end:
                  #<del>-</del>14, r7
                               ;r7=MLS-14
            add
171
                               ;r0=NLS = NLSmin + (MLS-14)
            add
                  r7, r0
i_i
            lds
                  r0, y0
35
13
      psha #16,y0
                               movs.w @r4+,x0
mi
                               movx.w @r4+,x1
            psha x0,y0,a0
                               movx.w @r4+,x0
            psha x1,y0,a1
                               movs.w a0,@r5+
 40
                               movx.w a1,@r5+
             psha x0,y0,a0
                               movx.w a0,@r5+
                               movx.w @r4+,x1
                               movx.w @r4+,x0
             psha x1,y0,a1
 45
                               movx.w a1,@r5+
             psha
                  x0,y0,a0
                               movx.w a0,@r5+
             rts
             nop
 50
```